(19) JAPAN PATENT OFFICE

PUBLICATION OF UNEXAMINED PATENT APPLICATION

(11) Publication number: JP-S50-92058

(43) Date of publication of application: July 23, S50 (1975)

(21) Application number: S48-141003

(22) Date of filing: December 12, S48 (1973)

Request for examination: Not requested

(Total 3pages)

Reference Numbers in the patent office: 6341 56

5825 56

(52) Japanese Cl.: 97(7)F112

97(7)C0

(51)Int. Cl.: G06F 9/20

G06F 13/00

PATENT APPLICATION (1)

December 12, Showa 48

To Commissioner, Patent Office

1. Title of the invention Information processing apparatus

2. Inventor

Address c/o Sanyo Electric Co Ltd

18-banchi, Keihanhondori 2-chome, Moriguchi-shi,

Name Suzuki Ryuji

3. Applicant

Address 18-banchi, Keihanhondori 2-chome, Moriguchi-shi

Name (188) Sanyo Electric Co Ltd

Representative Inoue Kaoru

4. List of Attached Materials

(1) Specification 1

(2) Drawings 1

(3) A duplicate of request 1

SPECIFICATION

1. TITLE OF THE INVENTION Information processing apparatus

2. RANGE OF THE PATENT CLAIMS

An information processing apparatus that executes instructions using a memory reference instruction that requires an execution address portion, the information processing apparatus, that is characterized composing a memory that stores words, word unit consisting of an operation instruction portion and an execution address portion; a memory address register into which an address for accessing the memory is introduced; a memory buffer register that temporarily stores a word that has been read out of the memory; an effective address control portion that performs a calculation of an effective address portion out of the word stored in the memory buffer register; and an operation control portion for executing the operation portion out of the word, wherein a format of the words stored in the memory has a construction where the effective address portion is corresponding to an address of an operation instruction portion of a next word, the calculation performed in the effective address control portion of the effective address portion of the word read out into the memory buffer register is for generating an address of an operation instruction portion of the next word, and the calculation of the effective address portion is completed before execution by the operation control portion.

3. DETAILED DESCRIPTION OF THE INVENTION

The present invention relates to an information processing apparatus, such as an electronic computer.

The instruction formats used in various kinds of information processing apparatuses are classified into memory reference instructions that require an effective address portion and non-memory reference instructions that do not require an effective address portion. The present invention provides an information processing apparatus that executes instructions using memory reference instructions out of these instruction formats.

In general, as shown in FIG. 1, the word format of memory reference instructions is composed of an operation instruction portion (OP) with a code showing +, -, ×, ÷, or the like, and an execution address portion (EA) that corresponds to the effective address, therefore, a program written using this kind of word format is shown in FIG. 2.

The time relations of the instruction execution written in this format is, the first, the effective address is found by calculating the effective address portion (EA), and then the content of the operation instruction portion (OP) is executed using the effective address found by the calculation. Accordingly, it is not possible to carry out the calculation of the effective address portion (EA) and the execution of the operation instruction portion (OP) simultaneously, and as a result, the time taken to execute the instruction is long.

The present invention was conceived in view of the above problem and is described in detail below.

The fundamental concept of the present invention is to construct the word format of memory reference instructions as shown in FIG. 3. That is, the operation instruction portion of the n^{th} word is the operation instruction portion (OP_n) of the n^{th} word, but the effective address portion is the effective address portion (EA_{n+1}) of the next word, that is, the $n+1^{th}$ word, in the same way, the operation instruction portion of the $n+1^{th}$ word is the operation instruction portion (OP_{n+1}) of the $n+1^{th}$ word, but being the effective address portion (EA_{n+2}) of the $n+1^{th}$ word.

The following describes the execution of instructions of a word format. FIG. 4 is a block diagram showing the construction, and (M) is a memory that stores words of the format described above, (MA) is a memory address register into which an address for accessing the memory is introduced, (MB) is a memory buffer register that temporarily stores a word that has been read from the memory (M), (EAC) is an effective address control portion for performing a calculation of an effective address portion (EA) out of the word stored in the memory buffer register (MB), (OPC) is an operation control portion for having the operation instruction portion (OP) in such word executed, (IR) is an instruction register, and (CTL) is a control signal source that issues a signal that has the both control portions (EAC), (OPC) perform specific operations in accordance with the content of the instruction register (IR). In addition, the effective address control portion (EAC) is composed of a register (RE) into which only the effective address portion (EA) out of the word stored in the memory buffer register (MB) is read out, an arithmetic circuit (AC) that calculates signals from the register (RE) and an effective address bus (EAB), an effective address register (EAR) into which a calculation result from the arithmetic circuit (AC) is

introduced, a program counter (PC), and an index register (ID). The program counter (PC) is used when the instruction format is relative addressing type, while the index register (ID) is used when being an index register type. The effective address register (EAR) and the program counter (PC) are connected via a memory address bus (MAB) to the memory address register (MA). The operation control portion (OPC) is constructed of an arithmetic logic circuit (ALu) into which only the operation instruction portion (OP) out of the word stored in the memory buffer register (MB) is read out and a plurality of arithmetic circuits (AC_1) (AC_2)..., with the arithmetic logic circuit (ALu) being connected to an input bus (IB) via an operation bus (OPB).

In the case of the n-1th word shown in FIG. 3, if the operation instruction portion (OP_{n-1}) is ignored for ease of explanation, the effective address portion (EA_n) is read out of from the memory buffer register (MB) into the register (RE) of the effective address control portion (EAC), the content of that is calculated with the content of the program counter (PC) or the index register (ID) by the arithmetic circuit (AC), and the result is stored in the effective address register (EAR). This calculation result is transferred to the memory address register (MA) via the memory address bus (MAB).

The memory (M) is accessed based on the address calculated by the effective address control portion (EAC) at the next nth word, and the accessed word is read out to the memory buffer register (MB) and the nth operation instruction portion (OP_n)with the calculated effective address is executed at the operation control portion (OPC). At the same time, the effective address portion (EA_{n+1}) of the nth word is calculated at the effective address control portion (EAC) in the same way as for the n-1th case and the result is stored in the effective address register (EAR).

After this, at the n+1th word, the n+1th operation instruction with the effective address calculated at the nth word is executed and the n+2th effective address is simultaneously calculated.

As a supplementary explanation for the method of the present invention, when for example the instruction of the nth word is "store the content of an accumulator in address 100 in the memory", the word format is as shown in FIG. 5, with address 100 for storing being written in the effective address portion (EA) of the nth word, a store instruction (ST) being written in the operation instruction portion (OP) of the n+1th word,

and the effective address portion (EA) being left blank.

When the operation instruction has a skip instruction, as shown in FIG. 6, a skip instruction (SKIP), which skips when a condition is satisfied but does not skip when the condition is not satisfied, is written into the operation instruction portion (OP) of the n^{th} word and the $n+1^{th}$ effective address (EA_{n+1}) is written into the effective address portion (EA), the $n+1^{th}$ operation (OP_{n+1}) is written in the operation instruction portion OP of the $n+1^{th}$ word, and the $n+2^{th}$ effective address (EA_{n+2}) is written into the effective address portion respectively. In the execution at the above structure, it is unclear whether the skip condition of the previous instruction is satisfied, when the skip condition is not satisfied, the operation instruction (OP_{n+1}) is executed using a value of the effective address (EA_{n+1}) and (EA_{n+2}) is simultaneously calculated, while on the other hand, when the skip condition is satisfied, the execution of the operation instruction (OP_{n+1}) is prohibited and only the calculation of (EA_{n+3}) is carried out.

As should be clear from the above explanation, with the present invention, the effective address of a next word is written into the effective address portion of a word and the effective address for an instruction operation is calculated in advance before the instruction operation is executed, so that the processing speed of various kinds of instructions is increased, therefore, the present invention makes a large contribution to information processing apparatuses where processing speed is instantly reflected in processing performance.

4. BRIEF DESCRIPTION OF THE DRAWINGS

FIGS. 1 and 2 are diagrams respectively showing the contents of conventional word formats, FIGS. 3, 5, and 6 are diagrams showing respective contents of word formats used by the apparatus of the present invention, and FIG. 4 is a block diagram showing the construction of an apparatus according of the present invention, (OP) designates an operation instruction portion, (EA) designates an effective address portion, (M) designates a memory, (MB) designates a memory buffer register, (EAC) designates an effective address control portion, and (OPC) designates an operation control portion.